[0175]   Attribute columns can be added to a query either by (a) selecting Attributes 1502 of the primary Entity class or (b) selecting Attributes 1503 of an Entity class that is related to the primary Entity class 1504. For example, if the primary Entity class 1501 is the Division Sections, then route type and average annual daily traffic can be part of the query because these are Attributes of the Division Sections. Similarly, bridge type can be part of the query because bridges are related to Division Sections (because they are co-located) and bridge type is an Attribute of the bridges class.

[0176]   Two types of filters can be used to either (a) select the primary Entities to include in the query result or, when the primary Entity class is Road Sections or Division Sections, (b) select the portions of the Entities to include in the query result. In either case, a filter can be based on Attributes of the primary Entity class or Attributes of a related Entity class.

[0177]   In the first case, the effect on the query result is obvious; the Entities (or, in case b, portions of the Entities) are restricted to those whose Attributes satisfy the filter constraint. For example, a query could report on the number of road-miles on state routes by filtering by the route type Attribute of the Division Section Entity class.

[0178]   In the second case, a set of related Entities is selected, and the relationship between the primary Entity class and the related Entity class selects primary Entities (or sections thereof) related to the selected Entities. For example, the number of miles of bridges could be determined by specifying Division Sections as

52

the primary Entity class and using the co-located relationship between bridges and Division Sections to select the portions of the roadway that are on bridges.

[0179]   Once the primary Entity class, the result Attributes, and the filters have been specified, the usual group by and order by operations can be specified to

5   complete the definition of the query. The following elements are part of creating a query: implementing table relationships; selecting primary Entity classes, and the types of queries that apply to each primary Entity class; including Attributes in queries, and the relationships used to join different Attributes in a single query result; filtering queries; grouping query results; creating columnar results; and converting

10   codes to labels. These elements are described below.

[0180]   An important concept for understanding and building queries is the relationship between the different types of tables used to store data. All data are stored as Attributes that are associated with an Entity class. Associated with each Entity class are some prototype table structures that are used to store data associated

15   with that Entity class. For the Entity classes as part of the road network data model, the tables used are exactly as defined. For the Entity classes defined as Entity Attributes data model, the tables are prototype tables that are "tuned" for each Entity class by altering table and column names and, in some instances, making minor changes to the table structure. The tuning process is controlled by tuning parameters

20   associated with the Entity/Attributes, and the parameters that specify how the tables for an Entity class have been tuned are stored in the Data Dictionary.

[0181]   The prototype tables defined in the road network data model and the

Entity Attributes data model define a small collection of table types. Thus, despite the

fact that the database may include hundreds of tables, the relationships between these

tables can be understood by defining the relationships that might exist between the

5   different pairs of table types. These relationships can be divided into two groups: (1)

relationships between Entity tables and Attribute tables and (2) predefined

relationships between Entity tables.

Predefined Entity-Attribute Relationships

10   [0182]   Associated with each Entity class is an Entity table that "defines" the

Entity IDs for the Entities in that Entity class; in other words, there is a one-to-one

correspondence between Entities in that Entity class and rows in the Entity table. For

the purpose of understanding this relationship, there are five types of Attributes: static

Attributes, Entity Attributes, dynseg (dynamic segment) Attributes, relate Attributes,

15   and lookup Attributes.

[0183]   A static Attribute is an Attribute that is stored as a column in the Entity

table of the parent Entity of that Attribute. Because most Entities allow for historical

information and an Entity table does not include historical columns, historical values

are not maintained for static Attributes. Also, static Attributes must be single-valued.

20   Because the Attribute is stored directly in the Entity table, no relationship information

is required to relate a static Attribute to its parent Entity.

54